UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/246,047	02/05/1999	EARL A. KILLIAN	83818-TEN-00	5721

7590 05/06/2002

ROGER S JOYNER
PILLSBURY MADISON & SUTRO
1100 NEW YORK AVENUE N W
NINTH FLOOR EAST TOWER
WASHINGTON, DC 200053918

EXAMINER
SIEK, VUTHE

ART UNIT	PAPER NUMBER
2825	

DATE MAILED: 05/06/2002

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/246,047	Applicant(s) KILLIAN ET AL.	
	Examiner Vuthe Siek	Art Unit 2825	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 February 2002* .

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-104* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-104* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

1. This office action is in response to application 09/246,047 and communication filed on 02/05/2002. Claims 1-104 remain pending in the application.

Claim Rejections - 35 USC § 112

1. Claim 2 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

2. Claim 2 recites "wherein the software development tools are for generating software development tools to generate code to run on the processor is not clear to what applicant intend to claim. First the software development tools are for generating software development tools is not understood and what code is it.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 1-21, 23-64 and 66-104 are rejected under 35 U.S.C. 102(b) as being anticipated by Hartoog et al, "GENERATION OF SOTWARE TOOLS FROM PROCESSOR DESCRIPTIONS FOR HARDWARE/SOFTWARE CODESIGN," IEEE, JUN 1997, PP. 303-306.

5. As to claims 1 and 104, Hartoog et al disclose a system and method for designing a configurable processor comprising means for, based on configurable

specification (single description of a processor in abstract), generating a description of a hardware implementation of the processor (nML is a high-level language at abstraction level the processor architecture, where the processor is a hardware); and means for, based on the configuration specification, generating software development tools specific to the hardware implementation (Tools suitable for designing the processor) (pages 303-306). Thus, the claimed invention is anticipated by Hartoog's reference.

6. As to claims 2-13, Hartoog et al disclose means for generating software development tools comprising: means for generating software development tools capable of generating code to run on the processor; a compiler; assembler; a linker; disassembler, debugger, wherein the debugger has a common interface and configuration set simulator, tailored to the configuration specification, for simulating code executable by the processor; instruction set simulator, wherein the instruction set simulator is capable of modeling execution of code being simulated the measure the key performance critical including cycles of execution, wherein the performance criteria are based on specific configuration microarchitectural features; wherein the instruction set simulator is capable of profiling execution of the program being simulated to record standard profiling statistics; wherein the hardware implementation description including a detailed HDL hardware implementation description (pages 303-305).

7. As to claims 14-20, Hartoog et al teach implementing hardware design from high level description language, where a netlist is generated in order to generate a physical design, where both software and hardware are verified. The above process inherently

includes synthesis scripts, programmable logic device scripts, a test bench, diagnostic tests for verification running on a simulator, and test tools (pages 303-305).

8. As to claim 21, Hartoog et al teach the means for generating the hardware implementation description comprising means for generating a hardware description language description; inherently including means for synthesizing logic and means for placing and routing (303-306)

9. As to claims 23-32, Hartoog et al teach means for generating the configuration specification in responsive to selection of configuration parameters, based on design goals for the processor, wherein the configuration specification including at least one parameter specification specifying the inclusion of a functional unit, and one processor instruction operating the function unit as Hartoog et al teach it could be reconfigured for bit width, number of registers, number of ALUs. ect. and parameter extensions (pages 303-305). Accordingly, reconfigured processor as taught by Hartoog et would have included a functional unit that is a multiplier, a multiply-accumulator unit, a digital signal processor (DSP processor on page 303) and a coprocessor in order to perform computation operation.

10. As to claims 33-43, Hartoog et al teach the processor description was a list of generic processor micro instructions to execute each target processor instruction, where the instructions were then executed on the generic VHDL model. It could be reconfigured for bit width, number of registers, number of ALUs, etc. Hartoog et al also teach parameter extensions (page 303-0-305). Thus, Hartoog et al teach one parameter specifies one of the inclusion, exclusion, and feature of a structure affecting

processor state, where the structure is a register file and the parameter specification specifies the number of registers; the structure is an instruction cache, wherein the parameter specification specifies the size of the cache, the line size of the cache, set associativity of the cache; wherein the structure is a data cache, wherein the parameter specifies the size of the cache, the line size of the cache, the set of associativity of the cache; the structure is a write buffer; the structure is one of an on-chip ROM and an on-chip RAM. In addition, above list of structure is well known in the design processor.

11. As to claims 44-46, Hartoog et al teach one parameter specifies a semantic characteristic, where the semantic characteristic is instruction byte ordering and the parameter specification specifies one of big endian and little endian byte ordering, wherein the semantic characteristic is a code density of the instruction set as Hartoog et al generating suite of software of tools as claimed that use the behavior from a different processor to pattern match and generate a different set of object code image (page 303-305). In addition, byte ordering and code density are well known within big endian and little endian conversion, where two different processors can be operated compatibly.

12. As to claims 47-54, Hartoog et al suggest one parameter specifies an execution characteristic controlling the execution of instruction in the processor, where the execution characteristic is known to be the number of external interrupts, the number of interrupt priority levels, the number of interrupt timers of the processor, wherein the execution characteristic is that a cache is one of a write-back cache and a write-through cache, wherein the execution characteristic is synchronization of the processor with

another processor, software controlled branch speculation, a windowing operation of processor registers (Fig. 1) (page 303-305, some features not mentioned in the article are inherently within design processors).

13. As to claims 55-59, Hartoog et al teach a suite of generated software tools include instruction set simulator, compiler instruction set simulator, disassembler, code generator, code retargeter and debugger (page 303-306). Hartoog et al teach given declarative description of an instruction set, it is possible to generate automatically several different useful tools above. Hartoog et al also it could be reconfigured for bit width, number of registers, number of ALUs, etc. in order to obtain a target processor Thus, Hartoog et al teach having at least one parameter specification specifies debugging characteristics of the processor, wherein the debugging characteristics include one of the inclusion and exclusion of an address trace and pipeline port, instruction and data break point support, verification support; and wherein the configuration specification include a parameter specification specifying at least one of selection of a predetermined feature; a size or number of a processor element; and an assignment of a value (pages 303-306).

14. As to claim 60-64, and 66-68, Hartoog et al teach means for evaluating suitability of the configuration specification since Hartoog et al teach generating a set of retargetable development tools for hardware and software codesign as claimed for a variety of processors, wherein the means for evaluating includes an interactive estimation tool, is for evaluating: hardware characteristics, the suitability of the configuration specification based on estimated performance characteristics, estimated

software characteristics of the processor further comprising means for providing information enabling modification of the configuration specification, and means for evaluating is for presenting a suitability evaluation to a user interactively by estimating at least one of code size and cycles (pages 303-306).

15. As to claims 69-71, Hartoog et al generating a good suite of development software tools from processor descriptions for a variety of processors for hardware/software codesign through a retargetable approach where the processor could be reconfigured for bit width, number of registers, number of ALUs etc. as claimed. Cost is well known to be used a metric to evaluating circuit design.

16. As to claims 72-91, Hartoog et al generating a good suite of development software tools from processor descriptions for a variety of processors for hardware/software codesign through a retargetable approach where the processor could be reconfigured for bit width, number of registers, number of ALUs etc. (pages 303-306). Given declarative description of an instruction set, it is automatically generated several different useful software tools for hardware/software codesign: instruction set simulator, compiler, disassembler, assembler, debugger, code generator and code retargeter for a variety of processors. Thus, Hartoog et al teach means for generating a configuration of the processor by extension (reconfigured processor including reconfigured for bit width, number of registers, number ALUs, ect.), wherein the extensible specification specifying an additional instruction, wherein the additional instruction adds no new state and adds state to the processor, inclusion of a user-defined instruction and implementation of the instruction, a new feature, and a

statement in instruction set architecture language specifying an opcode assignment and an instruction semantic, wherein the means for generating the software developments includes means for suggesting the user potential user-defined instructions particularly suited to at least on application, compiler, assembler, and simulator; and means for generating hardware implementation description is further for redefining and integrating the new feature into the detailed hardware implementation description, includes means for generating instruction decode logic, signals inherently specifying register operand usage for instruction interlock and stall logic (pages 303-306). Hartoog et al teach generating software development tools includes means for generating an instruction decode process, encode tables, and means for generating the hardware implementation description is further for generating a description of datapath hardware (pages 303-306).

17. As to claims 92-100, Hartoog et al generating a good suite of development software tools from processor descriptions for a variety of processors for hardware/software codesign through a retargetable approach where the processor could be reconfigured for bit width, number of registers, number of ALUs etc. (pages 303-306). Given declarative description of an instruction set, it is automatically generated several different useful software tools for hardware/software codesign: instruction set simulator, compiler, disassembler, assembler, debugger, code generator and code retargeter for a variety of processors. Thus, Hartoog et al teach the configuration specification includes at least a portion specified by an instruction set architecture description language description; means for generating the hardware

implementation description comprising means for generating instruction decode logic automatically, means for preprocessing for evaluating an expression and replacing the expression with a value; means for generating software development tools comprising means for generating an assembler, a compiler, a disassembler, an instruction set simulator, to modify the hardware implementation description; where the expression includes at least one of interactive construct, a conditional construct and inherently a database query in order to facilitate retargetable development tools.

18. As to claims 101-103, Hartoog et al generating a good suite of development software tools from processor descriptions for a variety of processors for hardware/software codesign through a retargetable approach where the processor could be reconfigured for bit width, number of registers, number of ALUs etc. (pages 303-306). Given declarative description of an instruction set, it is automatically generated several different useful software tools for hardware/software codesign: instruction set simulator, compiler, disassembler, assembler, debugger, code generator and code retargeter for a variety of processors. Thus, Hartoog et al teach the claimed invention of the configuration specification.

Claim Rejections - 35 USC § 103

19. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

20. Claims 22 and 65 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hartoog et al, "GENERATION OF SOTWARE TOOLS FROM PROCESSOR DESCRIPTIONS FOR HARDWARE/SOFTWARE CODESIGN," IEEE, JUN 1997, PP. 303-306.

21. As to claims 22 and 65, Hartoog et al suggest given this kind of declarative description of an instruction set, it is possible to generate automatically several different useful tools: Instruction Set Simulator, use the image to decode the instruction and then execute the behavior; Compiled Instruction Set Simulator, same as above, but decoding is done at model generating time to produce a simulation model for a processor running a specific program to decode each instruction every time it is executed (page 304, paragraph 3). Thus, Hartoog et al teach means for verifying timing and cycle time. It is well known to practitioner in circuit design that chip area and power dissipation are the main target. Although, Hartoog et al did explicitly mention in his article, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included means for determining the area and power dissipation of the circuit in order to minimize cost and enhance the circuit performance.

Remarks

22. Applicants agreed that Hartoog does generating software tools. However, Hartoog does not generate any hardware information, particularly a description of a hardware implementation of a processor. Examiner respectively submits that Hartoog et al teach that set of tools that generate instruction set simulators, assemblers, and disassemblers from a single description (common description) was developed to test if

retargettable development tools would work for commercial DSP processors and microprocessors. The processor instruction set was described using a language called nML. The nML is a high-level description language to describe the processor. Thus, the nML is generated for describing the hardware processor or is a description of a hardware implementation of a processor. Thus, Hartoog et al teach generating software tools and hardware from a common description. Examiner respectively submits that the nML language can be translated into hardware-description language such as VHDL or Verilog, wherein the VHDL or Verilog is supported by well known logic synthesis. Thus, HDL, synthesis scripts, place and route scripts are art inherent. In addition, examiner respectively submits that in order to design an IC including a processor, an IC specification should be documented before hand. In addition, examiner respectively submits that in order to design a processor it take more than two means as recited in the claims. The claimed invention is so broad. Based on interpretation of the claimed language and the reference, the current claims, especially, claims 1 and 104 are patentable over Hartoog's reference. The claimed invention recited, "means for", and applicants are requested to define specific means as recited in the claims. In addition, applicants are requested to clearly define what is the specific configuration specification as defined as recited in the claims. Also, applicants are requested to explain what "the software development tools are for generating software development tools to generate code to run on the processor" means.

2. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Conclusion

23. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Vuthe Siek whose telephone number is (703) 305-4958. The examiner can normally be reached on M-F (6:30-4:00) 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matthew Smith can be reached on (703) 308-1323. The fax phone numbers for the organization where this application or proceeding is assigned are (703) 305-3431 for regular communications and (703) 305-3431 for After Final communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 308-1782.

Vuthe Siek
Art Unit 2825
April 30, 2002